www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 5 may, 2014 Page No. 6216-6223

Cross lingual information retrieval using tf-idf and similarity measures

Prof. Pankaj Khambre, Aishwarya Pathak, Aarti Deshpande, Shringar Jha

GHRCEM , Pune(Maharashtra) India
pankaj.rick@gmail.com
It-Dept. , BVUCOEP,Pune(Maharashtra)India
Aishwarya0503@gmail.com
GHRCEM,Pune(Maharashtra)India
arti-scs@gmail.com
It-Dept. , BVUCOEP,Pune(Maharashtra)India
shringar.jha@gmail.com

Abstract:
This project demonstrates a simple and pragmatic approach for the creation of comparable corpora using Cross-Lingual Information Retrieval (CLIR). CLIR research is becoming more and more important for Information Retrieval (IR) on the Web as it is a truly multilingual environment and CLIR is necessary for global information exchange and knowledge sharing .In this project, the aim is to identify the same news story written in multiple languages (a problem of cross-language news story detection). For example, in a multilingual environment, such as India, where the same news story is covered in multiple languages, a reader might want to refer to the local language version of a news story and these are also rich sources of both parallel and comparable text. In the paper we have followed the corpus based approach for the retrieval of most relevant news.

Previous Work:

In recent years the development of internet and related topic has created a multilingual world-wide environment .In cross language IR either the documents or queries need to be translated .Research has concentrated on query translation rather than document translation because it is less expensive .Within the query translation framework basic approaches to CLIR are: Machine Translation (MT), Dictionary Based, Corpus Based.
The dictionaries used in CLIR are often bilingual machine readable dictionary (MRD).MRD translation uses a trivial method in which a source language word is replaced by all of its target equivalents ,all of which are taken to the final query .The basic problem are (1) Phrase translation. (2) Translation polysemy (translation ambiguity).If phrases are not identical, MRD translates phrases constitutes instead of full phrases and hence the sense of multiword keyword are lost .This results in decreased precision.

Basically, in dictionary based query translations, it is Dictionary based query translation is used for cross-lingual information retrieval. In this approach, a user enters his/her query in a language different from the language we

Have text collections in. That query is at first translated to the required language (generally English). This translation can be done using online dictionaries. In our approach, we have translated the target documents queries (English) to source documents language (Hindi). The translated text is then used as the new query, which goes as input to the Information Retrieval system. This retrieves the set of documents matching to that query.

Similarly, on a basic level, machine translation performs simple translations of words, from one natural language to another, but that alone cannot produce a good translation of text because recognition of whole phrase and their closest counterparts in the target language is needed. To solve such problem

usually corpus based technique is usually preferred because it leads to better translations and handling differences in linguistic topology, translation of idioms, and the isolation of anomalies.

Proposed Work:

In corpus based approach the basic translation is being done by both parallel corpora and comparable corpora. Basically parallel corpora are preferred due to its more accurate translation of knowledge but because of its scarcity comparable corpora are more preferred.

A comparable corpus is basically the one which selects the similar texts in more than one language or variety. For example, if we take an article from both Finnish and Swedish newspaper of same time period then we realise that a lot of topics from both the newspaper are similar. A comparable corpus could be created could be created by finding, for each article in the Finnish collection, a document in the Swedish collection that discusses the same article or event. Here, the Finnish collection is the source collection whereas the Swedish collection is the target collection.

In our work we have also proposed the use of tf idf algorithm.Tf–idf, short for term frequency–inverse document frequency, is a numerical statistic that reflects how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others.

1. Introduction

a) **Information retrieval** is finding material (usually documents) of an unstructured nature (usually text) that satisfies information need from within large collections (usually stored on computers). Searches can be based on metadata or on full-text indexing.Information retrieval systems are used to reduce what has been called information overload. Many universities and public libraries use IR systemsto provide access to books, journals and other documents. Web searchengines are the most visible IR applications. An information retrieval process begins when a user enters a query intothe system. Queries are formal statements of information needs, forexample search strings in web search engines. In information retrieval aquery does not uniquely identify a single object in the collection. Instead,several objects may match the query, perhaps with different degrees ofrelevance. An object is an entity that is represented by information in a database.User queries are matched against the database information. Depending on the data objects may be, for example, text documents, images, audio, or videos. Most IR systems compute a numeric score on how well each object in the database matches the query, and rank the objects according to this value. The top ranking objects are then shown to the user.

b) **CLIR** is a sub-field of information retrieval dealing with retrieving information written in a language different from the language of the user's query. For example,a user may pose their query in English but retrieve relevant documentswritten in French. To do so, most of CLIR systems use translation techniques.CLIR techniques can be classified into different categories based on different translation resources:

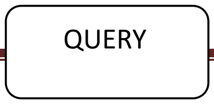
1. Dictionary-based CLIR techniques
2. Parallel corpora based CLIR techniques
3. Comparable corpora based CLIR techniques
4. Machine translator based CLIR techniques

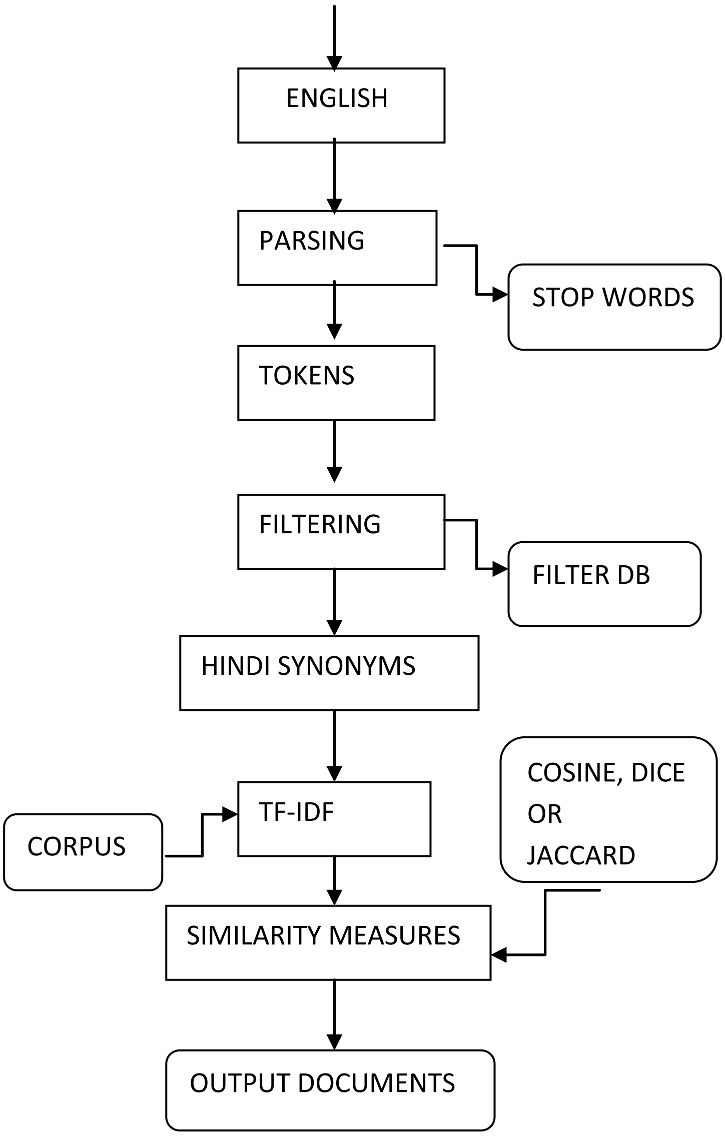
3. Query Input:

The search terms you enter and the order in which you enter them affect both the order and pages that appear in your search results. In the examples below, click on the similar ways of specifying various searches and note how the results differ. Query expansion (QE) is the process of reformulating a seed query to improve retrieval performance in information retrieval operations. In the context of web search engines, query expansion involves evaluating a user's input (what words were typed into the search query area, and sometimes other types of data) and expanding the search query to match additional documents. Query expansion involves techniques such as:

a) Finding synonyms of words, and searching for the synonyms as well
b) Finding all the various morphological forms of words by stemming each word in the search query
c) Fixing spelling errors and automatically searching for the corrected form or suggesting it in the results
d) Re-weighting the terms in the original query.

2. Flowchart:





4. Serialization and Tokenizer:

"Serialization is the process of converting the data (Objects) into stream of bytes and storing in to the files or database." The various steps for serialization are:

a) First let's create a new project in net beans. File ⇨ New Project.

b) Then create a Package which will help in managing the project. Right click on (Default Package) Add new ⇨ Java Package. Give an appropriate name to the package and click on Finish.

c) Now let's create a GUI form. Using which we

Let's

ck on

ive a

Now

ss for

Document consists of an optional headline or dateline followed by some text.

<Document><headline> | <dateline> | <textbegin>

<Textbegin><text_begin><word><text_end>

defining the field of data we will be storing for a single retrieval. As per our example we have three field news, date, and place. Let's create a class Library for this.

e) Now let's implement this class with interface "Serializable". After implementing now let's add the fields to this class.

Document consists of an optional headline or dateline followed by some text.

<Document><headline> | <dateline> | <textbegin>

<Textbegin><text_begin><word><text _end>

<Word><model> | <acronym> | <integer>, etc.

5. Parser:

Parsing is a procedure that recognizes a sentence and discover show it is built (i.e. gives its grammatical structure). Recognition involves finding out whether the sentence under consideration belongs to the particular language, i.e. whether it follows all the rules of well-formedness that the language prescribes. Discovering the structure involves identifying and marking the various components of a sentence - the phrases and the individual parts of speech such as noun, verb, preposition etc. Both the above functions require some concept of grammar of the underlying language. Parsing is the first step in natural language processing. Given a sentence, what is needed is a procedure that recognizes the sentence and also discovers how it is built. The execution of that procedure is called Parsing and the thing that executes it is called a Parser. This breakdown essentially is the first step in understanding the meaning of the sentence.

For document parsing, the grammar would be of the form as shown below. This shows that a Document may optionally consist of a headline, a dateline (the order is not restricted) and optionally some text. If the text occurs, it will consist of one or more words. Finally, a given word will either be a model, acronym, integer, or some other terminal.

Semantic parsing:

Semantic Parsing technique provides classification of linguisticentities into semantic types/role like Agent, Patient, and Instrument etc.

• Facilitates hierarchies to define sub- and super-types of concepts tomake relationship explicit (Hyponym-Hypernym Relation).e.g., Toyota and Ford are sub-types of cars, and Corolla and Carina aresub-types of Toyota

• Allow role structures to define components of entities. For example, the event ‘reading’ requires 2 roles: a reader and something to read(Selectional Restriction and Theta Roles).

• Accounts for syntactically correct but semantically nonsensicalsentence.e.g., Colorless green ideas sleep furiously

6. Vector Space Model.

I. Introduction

Efficient and effective text retrieval techniques are critical in managing the increasing amount of textual information available in electronic form. Yet text retrieval is a daunting task because it is difficult to extract the semantics of natural language texts. Most existing text retrieval technology rely on indexing keywords. Hence we use vector space model. Vector space model or term vector model is an algebraic model for representing text documents as vector or identifiers such as for e.g. index term.

II. Technique

Documents and queries are represented as vectors.

$$d_j = (w_{1,j}, w_{2,j}, \dots, w_{t,j})$$

$$q = (w_{1,q}, w_{2,q}, \dots, w_{t,q})$$

Each dimension corresponds to separate terms. If a term occurs in the document then its vector value is non zero.

Now we have a |v| dimensional space, where v is the number of terms in a document. The term, the word are the axes in the space. Documents are thought to be points or vectors in the space. The crucial properties of these vectors are that they are sparse vectors i.e. most of the terms here are 0.

Key idea 1: represent the queries as vectors in the space. **Key idea 2**: rank document according to their proximity to the query in space.

Proximity=similarity between vectors.

III. Formalizing vector proximity

The first step would be to cut distance between two points ie distance between two vectors. For this we can use Euclidean distances. But Euclidean distance may not be a good idea. This is because as seen in figure d1 has properties more relevant to 1 while d3 has properties more relevant to 2.The vector q is the one having properties of both. To get the relevance of d2 we can calculate the distance between vector q and d2.but this may not be efficient when we talk about large documents. So instead of calculating the distance, we can instead get the angle between them.

Key idea: rank documents according to the angle.

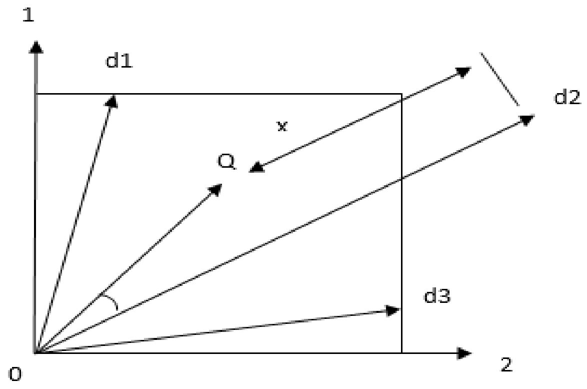


Figure: formalizing vector proximity

IV. Stages of vector space model

The vector space model can be divided basically into three stages. The first is document indexing where the content bearing terms are extracted from the document text. The second stage is weighting of the indexed terms to enhance retrieval of document relevant to user. The last stage would be the similarity measures.

6. TF-IDF Algorithm

Tf-idf stands for term frequency-inverse document frequency, and the tf-idf weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the tf-idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query.

Weight (t, D) = tf (t, D) * idf (t)

TF: Term Frequency, which measures how frequently a term occurs in a document. Since every document is different in length, it is possible that a term would appear much more times in long documents than shorter ones. Thus, the term frequency is often divided by the document length (aka. the total number of terms in the document) as a way of normalization

TF (t) = (Number of times term t appears in a document) / (Total number of terms in the document).

IDF: Inverse Document Frequency, which measures how important a term is. While computing TF, all terms are considered equally important. However it is known that certain terms, such as "is", "of", and "that", may appear a lot of times but have little importance. Thus we need to weigh down the frequent terms while scale up the rare ones, by computing the following:

IDF (t) = log_e (Total number of documents / Number of documents with term t in it).

Let:

D1=New York Times.

D2=New York Post.

D3=Los Angeles Times.

The total no. of documents in n=3.Therefore the **idf** values for the term:

Angles	Log(3/1)	1.548
Los	Log(3/1)	1.548
New	Log(3/2)	0.548
Post	Log(3/1)	1.548
Times	Log(3/2)	0.548
York	Log(3/2)	0.548

For all the documents, we calculate the **tf** score:

d/word	Angles	Los	New	Post	Times	York
D1	0	0	1	0	1	1
D2	0	0	1	1	0	1
D3	1	1	0	0	1	0

Hence the **tf-idf score** will be:

d/word	Angles	Los	New	Post	Times	York
D1	0	0	1.584	0	1.584	1.584
D2	0	0	1.584	1.584	0	1.584
D3	1.584	1.584	0	0	1.584	0

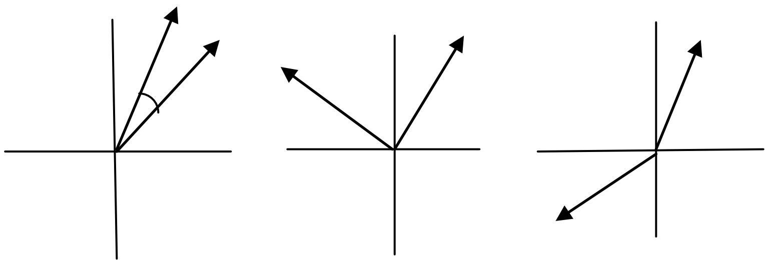
7. Similarity Measures (Cosine Formula)

This metric is frequently used when trying to determine similarity between two documents.In this similarity metric, the attributes (or words, in the case of the documents) is used as a vector to find the normalized dot product of the two documents. By determining the cosine similarity, the user is effectively trying to find cosine of the angle between the two objects.

$$\textbf{Similarity(x,y)= cos}(\theta)\textbf{=}$$

$$\frac{\mathbf{x\ dot\ y}}{||\mathbf{x}|| * ||\mathbf{y}||}$$

The resulting similarity ranges from −1 meaning exactly opposite, to 1 meaning exactly the same, with 0 usually indicating independence, and in-between values indicating intermediate similarity or dissimilarity.



1. Similar scores. Score vector in same direction. Angle between them is near 0 degree. Cosine of angle is near 1 i.e. 100 percent
2. Unrelated scores. Score vector are nearly orthogonal Angle between them is near 90 degree. Cosine of angle is near 0 i.e.00 percent

3. Opposite scores. Score vector in opposite direction. Angle between them is near 180 degree. Cosine of angle is near -1 i.e. -100 percent

Calculations:

Let's begin with the definition of dot product between two vectors, where An and Bn are the components of the vector (features of the document, or TF-IDF values for each word of the document in our example) and n is the dimensions of the vector.

$$\vec{A} = (A_1,A_2,A_3….A_N)$$

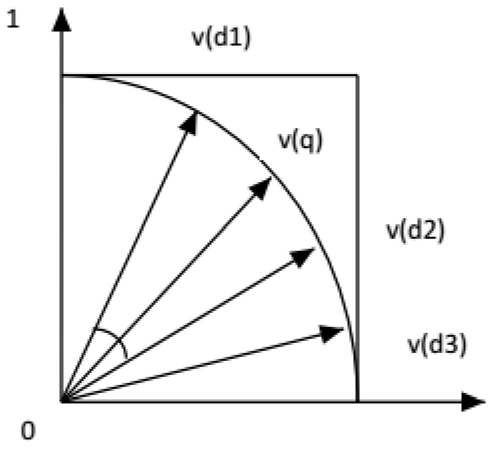
$$\vec{B} = (B_1,B_2,B_3…..B_N)$$

The cosine similarity between two vectors (or two documents on the Vector Space) is a measure that calculates the cosine of the angle between them.

This metric is a measurement of orientation and not magnitude; it can be seen as a comparison between documents on a normalized space because we're not taking into the consideration only the magnitude of each word count (tf-idf) of each document, but the angle between the documents. What we have to do to build the cosine similarity equation is to solve the equation of the dot product for the cos **(θ) =**

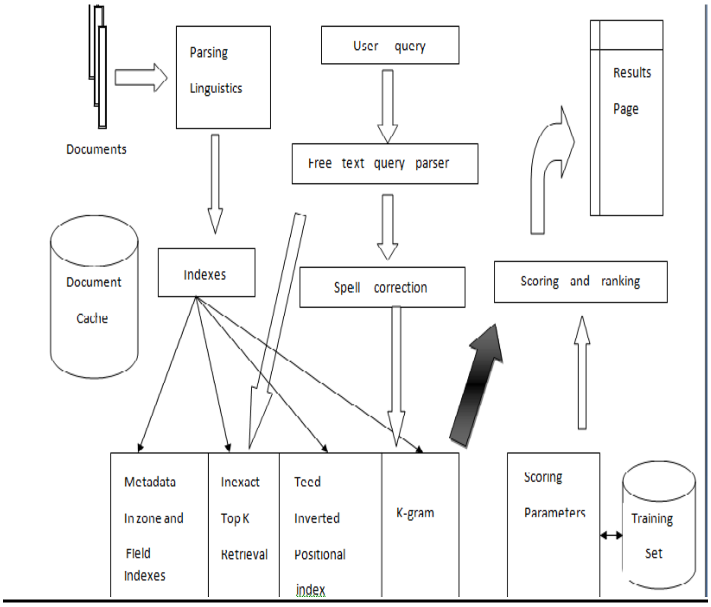
$$\frac{\mathbf{x\ dot\ y}}{||\mathbf{x}|| * ||\mathbf{y}||}$$

The figure below shows four vectors between the axes. The vector near to the axes name 1 will possess more characteristics of axes 1.the same will be the vector near to axes 2.
But the main question lies about the similarity of vector V (d2).it lie almost between 1 and 2. Hence the similarity of the vector can be calculated with the help of cosine similarity. The formula of which is the dot product between two products. The angle calculated will decide the similarity between the vectors. Here the vector may represent a query or a document.



Architectural Diagram:

A complete search system. Data diagrams are shown for a free text query.



8.Jaccard Index:

Jaccard coefficient is a common index for the binary variable. It is basically the quotient between the intersection and the union of the pair wise compared variables among the two objects. Generally, the information retrieval system uses the principle of words frequency that appears in the document with the weight of variable in the specified document and the proximity of user's request. Jaccard similarity coefficient is used to check the proximity of data in the process. It checks the proximity of the two data sets efficiently without the use of data redundancy. This method usually gives results of higher precision when a smaller database is used than a typical search page or otherwise. The search process starts by commencing users' queries to compare with the database. In case of input keyword matches with the index of the word in the database, those words can be accounted for the main keyword displayed in that search process. But if the query does not match the database, the process of similarity measurement can be proceeded to scrutinize the most similar words stored in the database.

$$\text{Jaccard sin (A, B)} = \frac{P(A\cap B)}{P(A\cup B)} \qquad \text{(A)}$$

The above mentioned formula is the most basic formula of Jaccard coefficient and hence, used for single or multiple keyword search in a short phase. A keyword search can be used effectively when similarity is computed within acceptance criteria.

Methodology:

In this research paper, there are two aspects which are need to be covered.

1) To find the information present for the keywords which are grammatically correct in the database.

2) To find the information on the keywords which are not grammatically correct i.e., misspelled words or over typed words.

1. **The association of two words in Jaccard coefficient:**
   Jaccard index is basically used to check the similarity, dissimilarity and the distance of data set. The basic association of words in **jaccard similarity coefficient** is measured by dividing the number of features that are common to all to the overall number of properties.

   $$\text{J (A, B)} = \frac{P(A\cap B)}{P(A\cup B)} \qquad \text{(B)}$$

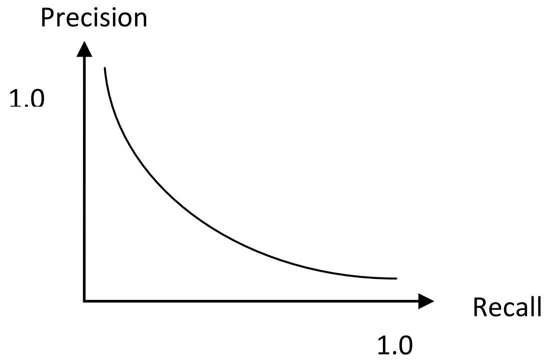
   Jaccard distance measures the dissimilarity between the sample sets and is complementary to the jaccard coefficient. Jaccard distance is a non similar measurement between data sets and is obtained by determining the inverse of jaccard coefficient. It is equal to the number of overall features minus number of features that are common to all, divided by the overall features.

   $$J_d\text{ (A,B)=1-J(A,B)} = \frac{(A\cup B) - (A\cap B)}{(A\cup B)} \qquad \text{(C)}$$

2. **The evaluation of the search words:**
   The evaluation of the searched words by using Jaccard similarity measure is tested by using following factors: precision, recall and F- measure.
   a) **Precision:**
   P= (Number of accurate results *100) / total number of answer retrieved by the system)

   b) **Recall:**
   R= (Number of accurate results*100) / total number of accurate results of raw data)

   c) **F- measure:**
   F= (2*Precision*Recall)/(Precision+Recall)



Precision Vs Recall Graph

Output:

From the output of implementation of the documented part shown in the research paper, we tally the result by saying that the probability for finding news or a noun in a document is highest for the cosine similarity than the jaccard.Degree of relevance shown bt vector space model may not be 100 percent but is highest as compared to other methods like jaccard and dice coefficient.

9. Conclusion and Further Work/Limitation

We can successfully conclude that manual translation of titles to form queries and then query based retrieval is the most accurate way of retrieval of comparable documents. Whatsoever, to make it more efficient, we use automated translation. Although it gives slightly inferior results in comparison to manual translation, it does not put extra burden of manually translating each and every title on the users. Hence we added a new aspect in the implementation. In our work we took the average of both the methods i.e. cosine measure and jaccard. The result being increased in the degree of relevance

Further work has to be done in the following fields:

1. **Multi-term phrases**: Failure to translate multi-term phrase is one of the reason for point decreased in the percentage of degree of relevance. In future we need to improvise the method or multi-term phrase.
2. **Source and target language:** We have successfully implementation English to Hindi translation. The same technique can be used for multiple languages.
3. **Proper nouns, Abbreviations and Compound work:** During outwork, we observed that dictionary often contained proper noun words. Like name of a town or a city. But the dictionary might not be well maintained to cover every inch of data about the place in the earth. Jargons or abbreviations are not translated and this affects the relevance factor in a document.

10. References

1. Accessing the compatibility of document, Emma Barker, Urand Rob in the department of computer science, University of Sheffield,2012.

2. Dragos Steffan Munteanu ,Daniel Marcu,Machine translation performance by using comparable corpora,2012

3. Christopher D Mannin,Prabhakar Raghvan,An introduction to information retrieval Cambridge University,2009

4. Lice Ballestros,Bruce Croft, Dictionary method for information retrieval l998.